

AMENDMENTS

In the Claims:

Please amend the claims as indicated hereafter.

1. (Previously Presented) A computer system for processing instructions of a computer program, comprising:
 - a plurality of registers;
 - a plurality of connections corresponding respectively with said registers;
 - at least one pipeline configured to process and execute said instructions;
 - a scoreboard coupled to said plurality of connections and to said at least one pipeline, said scoreboard having a plurality of bits corresponding respectively with said plurality of registers, said scoreboard configured to transmit each of said bits across a different one of said connections, each of said bits indicative of whether a pending write to a corresponding one of said registers exists;
 - decoding circuitry configured to decode at least one encoded register identifier associated with at least one of said instructions into a decoded register identifier, said decoded register identifier having a plurality of bits corresponding respectively with said plurality of registers and identifying at least one of said registers; and
 - hazard detection circuitry coupled to each of said plurality of connections, said hazard detection circuitry configured to compare each of said transmitted bits to a respective one of said bits of said decoded register identifier and to detect data hazards based on comparisons of said transmitted bits to said decoded register identifier bits.

2. (Previously Presented) The system of claim 1, wherein said transmitted bits form a data word transmitted from said scoreboard to said hazard detection circuitry, each asserted bit in said data word indicating that a different one of said registers is associated with a pending write.

3. (Previously Presented) The system of claim 1, wherein said scoreboard includes a plurality of registers, each of said scoreboard registers containing a different one of said scoreboard bits and connected to a different one of said connections.

4-5. (Canceled).

6. (Previously Presented) The system of claim 1, wherein said decoded register identifier includes a plurality of bits, wherein each of said bits in said decoded register identifier that corresponds to said at least one register is asserted, and wherein the remaining bits in said decoded register identifier are deasserted.

7. (Previously Presented) A system for processing instructions of computer programs, comprising:

at least one pipeline;

a plurality of registers;

a plurality of connections, each of said connections corresponding to a different one of said registers;

means for maintaining a plurality of bits and for indicating via said bits which of said registers is associated with a pending write, said maintaining means configured to transmit said bits across said connections, wherein each bit transmitted across each of said connections is indicative of whether the register corresponding to said each connection is associated with a pending write; and

hazard detection circuitry configured to perform comparisons between said bits and a decoded register identifier associated with at least one instruction presently in said at least one pipeline, said hazard detection circuitry further configured to detect at least one data hazard based on said comparisons.

8. (Previously Presented) The system of claim 7, wherein said maintaining means includes a plurality of registers, each of said registers of said maintaining means containing a different one of said bits and connected to a different one of said connections.

9. (Previously Presented) The system of claim 7, further comprising means for decoding, into said decoded register identifier, an encoded register identifier associated with said at least one instruction, said decoding means configured to transmit said decoded register identifier to said hazard detection circuitry and to said maintaining means, said decoded register identifier identifying at least one of said registers.

10. (Previously Presented) The system of claim 9, wherein said decoded register identifier includes a plurality of bits, wherein each of said bits in said decoded register identifier that corresponds to said at least one register is asserted, and wherein the remaining of said bits in said decoded register identifier are deasserted.

11. (Previously Presented) A method for processing instructions of computer programs, comprising the steps of:

processing said instructions via at least one pipeline;

providing a plurality of registers;

maintaining a plurality of bits, each of said bits indicating whether a corresponding one of said registers is associated with a pending write;

transmitting a data word, said data word including each of said bits, wherein each asserted bit in said data word indicates that a different one of said registers is associated with a pending write;

receiving said data word;

comparing said data word to a decoded register identifier associated with at least one instruction presently in said at least one pipeline; and

detecting a data hazard based on said comparing step.

12. (Previously Presented) The method of claim 11, further comprising the step of modifying one of said bits based on said decoded register identifier.

13. (Previously Presented) A method for processing instructions of computer programs, comprising the steps of:

processing said instructions via at least one pipeline;

providing a plurality of registers;

maintaining a plurality of bits within a scoreboard, each of said bits respectively corresponding with one of said registers;

providing a plurality of connections, each of said connections respectively corresponding with one of said registers;

indicating, via said bits, which of said registers are associated with pending writes;

transmitting, from said scoreboard, each of said bits across a different one of said connections;

decoding at least one register identifier associated with at least one of said instructions into a decoded register identifier, said decoded register identifier having a plurality of bits corresponding respectively with said plurality of registers and identifying at least one of said registers;

comparing each of said transmitted bits to a respective one of said bits of said decoded register identifier; and

detecting a data hazard based on said comparing step.

14. (Canceled).

15. (Previously Presented) The method of claim 13, further comprising the step of modifying one of said scoreboard bits based on said decoded register identifier.

16. (Previously Presented) A system for processing instructions of computer programs, comprising:

a plurality of registers;

a plurality of connections;

at least one pipeline;

a scoreboard having data indicative of whether each of said plurality of registers is respectively associated with a pending write, said scoreboard configured to transmit said data across said connections; and

hazard detection circuitry coupled to said connections and configured to receive said transmitted data and to perform a comparison between said transmitted data and a decoded register identifier associated with an instruction presently in said at least one pipeline, said hazard detection circuitry further configured to detect a data hazard based on said comparison.

17. (Previously Presented) The system of claim 16, further comprising a decoder configured to receive an encoded register identifier associated with said instruction, said decoder further configured to decode said encoded register identifier into said decoded register identifier and to transmit said decoded register identifier to said hazard detection circuitry and to said scoreboard.

18. (Currently Amended) The system of claim 17, wherein said transmitted data comprises a plurality of bits, each of said bits indicative of whether a corresponding one of said registers is associated with a pending [[write]] write.

19. (Previously Presented) A method for processing instructions of computer programs, comprising the steps of:

processing said instructions via at least one pipeline;

using a plurality of registers to execute said instructions;

storing, in a scoreboard, data indicative of which of said registers is associated with a pending write;

transmitting said data from said scoreboard;

decoding an encoded register identifier associated with one of said instructions;

comparing said transmitted data to said decoded register identifier; and

detecting a data hazard based on said comparing step.

20. (Previously Presented) The method of claim 19, wherein said transmitted data comprises a plurality of bits, each of said bits indicative of whether a corresponding one of said registers is associated with a pending write.

21. (Previously Presented) The method of claim 19, further comprising the steps of:

transmitting said decoded register identifier to said scoreboard; and

modifying said data based on said decoded register identifier.